IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Applicant:	§	
	§	
MICHAEL KAGAN ET AL.	§	
	§	
Serial No.: 10/052,500	§	
	§	
Filed: January 23, 2002	§	Group Art Unit: 2157
	§	
For: DOORBELL HANDLING WITH PRIORITY PROCESSING FUNCTION	§	Attorney Docket: 3091/20
	§	
	§	
Examiner: Sargon N. Nano	§	

TRANSMITTAL OF APPEAL BRIEF

Commissioner of Patents and Trademarks
Washington, DC 20231

Dear Sir:

Transmitted herewith in triplicate is the APPEAL BRIEF in this application with respect to the Notice of Appeal filed on July 24, 2006.

The application is on behalf of

___ other than a small entity

_X_ small entity

verified statement:

___ attached

_X_ already filed

Pursuant to 37 CFR 1.17(f) the fee for filing the Appeal Brief is:

_X_	small entity	$ 250
___	other than a small entity	$ 500

Appeal Brief fee due $250

_X_ Applicant petitions for an extension of time under 37 CFR 1.136 for the total number of months checked below:

	small entity	not small entity
___one month	$ 60	$ 120
___two months	$ 225	$ 450
_X_three months	$ 510	$ 1020
___four months	$ 795	$ 1590

If an additional extension of time is required please consider this a petition therefor.

The total fee due is:

Appeal brief	$250
Extension fee (if any)	$510
TOTAL FEE DUE	$760

Please charge Account No. 06-2140 the sum of $760. A duplicate copy of this transmittal letter is attached.

If any additional extension and/or fee is required, this is a request therefor and to charge Account No. 06-2140.

If any additional fee for claims is required, please charge Account No. 06-2140.

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Applicant:	§	
	§	
MICHAEL KAGAN	§	
	§	
Serial No.: 10/052,500	§	
	§	
Filed: January 23, 2002	§	Group Art Unit: 2157
	§	
For: DOORBELL HANDLING WITH PRIORITY PROCESSING FUNCTION	§	Attorney Docket: 3091/20
	§	
	§	
Examiner: Sargon N. Nano	§	

Commissioner of Patents and Trademarks
Washington, DC 20231
ATTENTION: Board of Patent Appeals and Interferences

APPELLANT'S BRIEF

Dear Sir:

This is in furtherance of the Notice of Appeal filed in this case on July 24, 2006.

The fees required under § 1.17(f) and any required petition for extension of time for filing this brief and fees therefor are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate.

This brief contains these items under the following headings and in the order set forth below:

I. REAL PARTY IN INTEREST

II. RELATED APPEALS AND INTERFERENCES

III. STATUS OF CLAIMS

IV. STATUS OF AMENDMENTS

V. SUMMARY OF CLAIMED SUBJECT MATTER

VI. ISSUES

VII. ARGUMENTS

_X_ ARGUMENT: VIIA REJECTIONS UNDER 35 U.S.C. 102

VIII. APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

IX. APPENDIX OF EVIDENCE

X. APPENDIX OF RELATED PROCEEDINGS

I. REAL PARTY IN INTEREST

The real party in interest in this case is:

Mellanox Technologies, Ltd.

P. O. Box 586

20692 Yokneam

ISRAEL

II. RELATED APPEALS AND INTERFERENCES

NONE

III. STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-29

B. STATUS OF ALL THE CLAIMS

1. Claims cancelled: NONE
2. Claims withdrawn from consideration but not cancelled: NONE
3. Claims pending: 1-29
4. Claims allowed: NONE
5. Claims rejected: 1-29

C. CLAIMS ON APPEAL

The claims on appeal are: claims 1-29.

IV. STATUS OF AMENDMENTS

No claims were amended subsequent to the final rejection.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 recites a method for communication over a network (Figure 1 network 26). One or more doorbell addresses (Figure 1 doorbells 36; page 2 lines 1-4; page 4 lines 12-15; page 16 lines 7-13) are assigned on a network interface adapter (Figure 1 HCA 22) for use by a host processor (Figure 1 CPU 24). A first descriptor (Figure 1 descriptor 34), that defines a first message to be sent over the network (page 1 lines 15-17; page 4 lines 8-9; page 15 lines 24-26), is written to a system memory (Figure 1 memory 32) associated with the host processor. A command is written to a first doorbell address (page 1 line 28 to page 2 line 1; page 4 lines 9-12; page 5 lines 15-17; page 16 lines 3-4) instructing the network interface adapter to read and execute the first descriptor. A second descriptor is written to a second doorbell address to define a second message to be sent over the network (page 5 lines 17-19; page 17 lines 1-4). Responsive to the command having been written to the first doorbell address, the first descriptor is read from the system memory using the network interface adapter (page 16 lines 18-20; page 19 lines 12-15; Figure 3 step 74). Responsive to the first descriptor, the first message is sent from the network interface adapter over the network (page 19 lines 19-25; Figure 3 step 80). Responsive to the second descriptor having been written to the second doorbell address (page 17 lines 23-26; page 19 lines 15-19, Figure 3 steps 70 and 76), the second message is sent from the network interface adapter over the network (page 19 lines 19-25; Figure 3 step 80).

Independent claim 10 recites a method for direct memory access. A first descriptor (Figure 1 descriptor 34), that defines a first operation (page 1 lines 15-17; page 4 lines 8-9) for execution by a DMA engine (Figure 1 HCA 22), is written to a system memory (Figure 1 memory 32) associated with a host processor (Figure 1

CPU 24). A command is written to a first doorbell address (page 1 line 28 to page 2 line 1; page 4 lines 9-12; page 5 lines 15-17; page 16 lines 3-4) of the DMA engine instructing the DMA engine to read and execute the first descriptor. A second descriptor is written to a second doorbell address of the DMA engine to define a second operation for execution by the DMA engine (page 5 lines 17-19; page 17 lines 1-4). Responsive to the command having been written to the first doorbell address, the first descriptor is read from the system memory and is executed, using the DMA engine (page 16 lines 18-20; page 19 lines 12-15; Figure 3 step 74). Responsive to the second descriptor having been written to the second doorbell address, the second descriptor is executed using the DMA engine (page 17 lines 23-26; page 19 lines 15-25; Figure 3 steps 70, 76 and 80).

Independent claim 15 recites a network interface adapter (Figure 1 HCA 22), for coupling to a host processor (Figure 1 CPU 24), that comprises a range of doorbell addresses, execution circuitry, and a doorbell handler. The doorbell addresses (Figure 1 doorbells 36), including first and second doorbell addresses, are in an address space of the host processor (page 2 lines 1-4; page 4 lines 12-15; page 16 lines 7-13). The execution circuitry (Figure 2 execution unit 52 and gather engine 54) sends messages over the network (Figure 1 network 26) responsive to descriptors, including first and second descriptors, that are prepared by the host processor (page 19 lines 12-25).

The doorbell handler (Figure 2 DB preprocessor 42, priority buffer 46 and DB handler 48) is coupled to the range of doorbell addresses so as to receive a command written by the host processor to the first doorbell address (page 5 lines 15-17; page 18 lines 13-15) and so as to receive the second descriptor written by the host processor to the second doorbell address (page 5 lines 17-19; page 17 lines 23-26). The command indicates that the first descriptor (Figure 1, descriptor 34), that defines a first message

(page 2 lines 13-14; page 15 lines 24-26), has been written to a system memory (Figure 1 memory 32) associated with the host processor. The second descriptor defines a second message (page 2 lines 13-14). Responsive to the command having been written to the first doorbell address (page 17 lines 7-10), the doorbell handler instructs the execution circuitry to read the first descriptor from system memory and to execute the first descriptor so as to send the first message (page 17 lines 14-22). Responsive to the second descriptor having been written to the second doorbell address (page 17 lines 23-26), the doorbell handler passes the second descriptor to the execution circuitry to execute the second descriptor so as to send the second message (page 19 lines 15-19).

Independent claim 24 recites a host channel adapter (Figure 1, HCA 22), for coupling a host processor (Figure 1, CPU 24) to a switch fabric (Figure 1, network 26), that comprises a range of doorbell addresses, execution circuitry and a doorbell handler. The doorbell addresses (Figure 1 doorbells 36), including first and second doorbell addresses, are in an address space of the host processor (page 2 lines 1-4; page 4 liines 12-15; page 16 lines 7-13). The execution circuitry (Figure 2 execution unit 52 and gather engine 54) generates data packets for transmission over the network (page 19 lines 12-25) responsive to work requests (page 4 line 8: work requests are descriptors), including first and second work requests.

The doorbell handler (Figure 2 DB preprocessor 42, priority buffer 46 and DB handler 48) is coupled to the range of doorbell addresses so as to receive a command written by the host processor to the first doorbell address (page 5 lines 15-17; page 18 lines 13-15) and so as to receive the second work request written by the host processor to the second doorbell address (page 5 lines 17-19; page 17 lines 23-26). The command indicates that the first work request (Figure 1, descriptor 34) has been

written to a system memory (Figure 1 memory 32) associated with the host processor. Responsive to the command having been written to the first doorbell address (page 17 lines 7-10), the doorbell handler passes instructions to the execution circuitry (page 17 lines 14-22; page 18 lines 12-15) to read the first work request from system memory (page 19 lines 14-15) and to execute a first work queue element corresponding to the first work request (page 3 line 31 through page 4 line 2) so as to generate the data packets called for by the first work request (page 19 lines 19-25). Responsive to the second work request having been written to the second doorbell address (page 17 lines 23-26), the doorbell handler passes a work queue element corresponding to the second work request to the execution circuitry (page 19 lines 15-19) and instructs the exeution circuitry to execute the second work queue element so as to generate the data packets called for by the second work request (page 19 lines 19-25).

Independent claim 25 recites a direct memory access device that comprises a range of doorbell addresses, a DMA engine and a doorbell handler. The doorbell addresses (Figure 1 doorbells 36), including first and second doorbell addresses, are in an address space of a host processor (Figure 1 CPU 24; page 2 lines 1-4; page 4 liines 12-15; page 16 lines 7-13). The DMA engine (Figure 2 execution unit 52 and gather engine 54), responsive to descriptors prepared by the host processor (page 1 lines 25-28; page 15 lines 24-26), accesses a system memory (Figure 1 memory 32; page 19 lines 22-23) associated with the host processor. The descriptors include first and second descriptors that define first and second operations for execution by the DMA engine (page 1 liines 15-17).

The doorbell handler (Figure 2 DB preprocessor 42, priority buffer 46 and DB handler 48) is coupled to the range of doorbell addresses so as to receive a command written by the host processor to the first doorbell address indicating that the first

descriptor (Figure 1 descriptor 34) has been written to system memory (page 5 lines 15-17; page 18 lines 13-15) and so as to receive the second descriptor written by the host processor to the second doorbell address (page 5 lines 17-19; page 17 lines 23-26). Responsive to the command having been written to the first doorbell address (page 17 lines 7-10), the doorbell handler instructs the DMA engine to execute the first operation responsive to the first descriptor from system memory (page 17 lines 14-22). Responsive to the second descriptor having been written to the second doorbell address (page 17 lines 23-26), the doorbell handler instructs the DMA engine to execute the second operation (page 19 lines 15-19).

VI. ISSUES

Whether claims 1-29 are anticipated by Gronke, US Patent No. 6,888,792 (henceforth, "Gronke '792").

VII. ARGUMENTS

VIIA ARGUMENTS - REJECTIONS UNDER 35 U.S.C. 102

In a high-speed, packetized, serial input/output network architecture such as InfiniBand™, a host is coupled to the network by a network adapter. The memory space of the adapter includes "doorbells" (typically one page each in size) that are used by applications running on the host to notify the adapter that data in host memory are to be sent to other entities on the network. According to the prior art, when an application running on the host needs to send data from host memory to another entity on the network, the application writes a descriptor of the data to host memory and then "rings" a "doorbell" by writing an appropriate command to the doorbell to inform the adapter of the location of the descriptor in host memory. In response to the command, the adapter reads the descriptor and performs a corresponding direct memory access to fetch the data.

Gronke '792 teaches a method by which nodes communicate with each other over a plurality of fabrics. Each node has a set of physical ports associated with each fabric and a map that associates physical ports with virtual ports. Each node includes a network interface controller (NIC) **18**, for example a host channel adapter (HCA) **210** or a target channel adapter (TCA) **242**, that associates the node's data transfer operations with respective virtual ports. When a fabric fails, NIC **18** remaps the virtual ports associated with the failed fabric to virtual ports that are associated with functioning fabrics.

The node notifies NIC **18** of direct memory access data transfer operations in only one way: the conventional manner described above. The node posts descriptors **23** of a data transfer operation in a send queue **21** or in a receive queue **19** and then

rings the associated doorbell **25** or **27**. This is stated explicitly in Gronke '792 in column 3 lines 17-22:

> A send doorbell **25** and a receive doorbell **27** are provided for allowing the VI consumer to notify the VI NIC **18** that work has been placed in the send queue **19** and receive queue **21**, respectively, meaning that a descriptor describing a requested data transfer operation has been placed in queues **19** and **21**. (emphasis added)

and in column 5 lines 50-56:

> According to an example embodiment, the host node...places descriptors into send queues for a send operation or into receive queues for a receive operation, and then rings a doorbell to notify the HCA that work has been placed in the work queues. The HCA then sends or receives the data over a channel. (emphasis added)

Consequently, NIC **18** fetches the data in only one way: in response to being notified by the ringing of a doorbell that a descriptor **23** has been posted in a send queue **21** or in a receive queue **19**. This is stated explicitly in Gronke '792 in column 4 lines 2-7:

> A consumer **8** posts descriptors, or places the descriptors in a work queue then rings a doorbell to notify the NIC that work has been placed in the work queue. The VI NIC **18** then processes the descriptor by sending or receiving data (directly between application memory and network without kernel processing)...

That Gronke '792 performs conventional direct memory access also is evident from Figures 1B through 4. Figure 2 shows send queue **214** and receive queue **216** of host **202** as separate from HCA **210** of host **202** and shows send queue **224** and receive queue **226** of host **220** as separate from HCA **222** of host **220**. The only place in host **202**, as illustrated in Figure 3, that send queue **214** and receive queue **216** could be is in main memory **206**. Similarly, the only place in host **220** that send queue **224** and receive queue **226** could be is in a main memory of host **220** that is separate from the HCA of host **220**. Returning to Figure 2, this Figure shows send queue **244** and receive queue **246** of I/O unit **240** as separate from TCA **242** of I/O

unit **240**. The only place in I/O unit **240**, as illustrated in Figure 4, that send queue **244** and receive queue **246** could be is in main memory **415**. Therefore, the descriptors **23** shown in send queue **19** and in receive queue **21** of Figure 1B must be stored in a main memory of the node illustrated in Figure 1B, and not in doorbell of a network interface controller.

By contrast the present invention provides two ways to notify a NIC or its equivalent of direct memory access data transfer operations. The first way is the conventional one used by Gronke '792. The second way is to ring a doorbell by writing descriptors directly to the doorbell. (The purpose of the second way is to give the associated data transfer operations priority over the data transfer operations that are handled in the conventional manner.) Similarly, the present invention provides two corresponding ways for the NIC or its equivalen to perform the direct memory access data transfer operations. The first way is the conventional one used by Gronke '792. The second way is to respond to the descriptors having been written directly to the doorbell by performing the corresponding direct memory access data transfer operations. These aspects of the present invention are recited in the claims by using two different ways to recite the first and second ways of notifying a NIC or its equivalent of data transfer operations, and by using two different ways to recite the two corresponding ways of performing of the data transfer operations by the NIC or its equivalent.

The first way of notifying a NIC or its equivalent of a data transfer operation and the first way of peforming the data transfer operation by the NIC or its equivalent are recited in clam 1 as the steps of:

> writing a first descriptor to a system memory...
> writing a command to a firs one of the doorbell addresses...

responsive to the command having been written to the first one of the doorbell addresses, reading the first descriptor from the system memory...

in claim 10 as the steps of:

writing a first descriptor to a system memory...
writing a command to a first doorbell address of the DMA engine...
responsive to the command written to the first doorbell address, reading the first descriptor from the system memory...

in claim 15 as follows:

a doorbell handler, which is coupled to the range of doorbell addresses so as to receive a command written...to the first doorbell address, indicating that the first descriptor has been written to a system memory...the doorbell handler being further coupled, responsive to the command having been written to the first doorbell address, to instruct the execution circuitry to read the first descriptor from the system memory and to execute the first descriptor...

in claim 24 as follows:

a doorbell handler, which is coupled to the range of doorbell addresses so as to receive a command written...to the first doorbell address, indicating that the first work request has been written to a system memory...the doorbell handler being further coupled, responsive to the command having been written to the first doorbell address, to pass instructions to the execution circuitry to read the first work request from the system memory and to execute a first work queue element corresponding to the first work request...

and in claim 25 as follows:

a doorbell handler, which is coupled to the range of doorbell addresses so as to receive a command...to the first doorbell address, indicating that the first descriptor has been written to the system memory...the doorbell handler being further coupled, responsive to the command having been written to the first doorbell address, to instruct the DMA engine to execute the first operation responsive to the first descriptor in the system memory...

The second way of notifying a NIC or its equivalent of a data transfer operation and the second way of performing the data transfer operation by the NIC or its equivalent are recited in clam 1 as the steps of:

writing a second descriptor to a second one of the doorbell addresses...

responsive to the second descriptor having been written to the second one of the doorbell addresses, sending the second message from the network interface adapter over the network.

in claim 10 as the steps of:

writing a second descriptor to a second doorbell address of the DMA engine...

responsive to the second descriptor having been written to the second doorbell address, executing the second descriptor using the DMA engine.

in claim 15 as follows:

a doorbell handler, which is coupled to the range of doorbell addresses...so as to receive the second descriptor written...to the second doorbell address...the doorbell handler being further coupled...responsive to the second descriptor having been written to the second doorbell address, to pass the second descriptor to the execution circuitry and to instruct the execution circuitry to execute the second descriptor...

in claim 24 as follows:

a doorbell handler, which is coupled to the range of doorbell addresses...so as to receive the second work request written by the host processor to the second doorbell address, the doorbell handler being further coupled...responsive to the second work request having been written to the second doorbell address, to pass a work queue element corresponding to the second work request to the execution circuitry to execute the second work queue element...

and in claim 25 as follows:

a doorbell handler, which is coupled to the range of doorbell addresses...so as to receive the second descriptor written by the host processor to the second doorbell address, the doorbell handler being further coupled...responsive to the second descriptor having been written to the second doorbell address, to instruct the DMA engine to execute the second operation.

Therefore, the teachings of Gronke '792 have absolutely nothing to do with the present invention. Gronke '792 teaches the claim limitations associated with the conventional method of direct memory access but not the innovative claim limitatioins associated with the present invention's second method of direct memory

access. There is neither a hint nor a suggestion in Gronke '792 of having two ways to ring a doorbell and then perform dirrect memory access in response to the ringing of the doorbell. Therefore, independent claims 1, 10, 15, 24 and 25 are allowable in their present form. With independent claims 1, 10, 15 and 25 allowable in their present form, it follows that claims 2-9, 11-14, 16-23 and 26-29 that depend therefrom also are allowable.

The second method of direct memory access also was described, in the response to the office action mailed September 8, 2005, as "writing descriptors directly to the doorbell". What was meant then, and what is meant above, by writing "directly" to the doorbell is as opposed to writing the descriptors to host memory and then notifying the adaptor, by ringing a doorbell, that the descriptors are available for reading in host memory. In the office action mailed February 23, 2006, the Examiner interpreted Applicant's argument as follows:

> Applicant is arguing that Gronke does not disclose "a second way to ring a doorbell by writing descriptions directly to the doorbell". This/These limitation(s) are not found in the claims.

Applicant submits that this limitation is in fact found in the claims, as follows:

> Claim 1:
>
> writing a second descriptor to a second one of the doorbell addresses...
>
> Claim 10:
>
> writing a second descriptor to a second doorbell address of the DMA engine...
>
> Claim 15:
>
> ...so as to receive the second descriptor written by the host processor to the second doorbell address...
>
> Claim 24:
>
> ...so as to receive the second work request written by the host processor to the second doorbell address...

Claim 25:

...so as to receive the second descriptor written by the host processor to the second doorbell address...

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: February 4, 2007

VIII. APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

The text of the claims on appeal is:

1. A method for communication over a network, comprising:

assigning one or more doorbell addresses on a network interface adapter for use by a host processor;

writing a first descriptor to a system memory associated with the host processor, the first descriptor defining a first message to be sent over the network;

writing a command to a first one of the doorbell addresses instructing the adapter to read and execute the first descriptor;

writing a second descriptor to a second one of the doorbell addresses, the second descriptor defining a second message to be sent over the network;

responsive to the command having been written to the first one of the doorbell addresses, reading the first descriptor from the system memory using the network interface adapter, and sending the first message from the network interface adapter over the network responsive to the first descriptor; and

responsive to the second descriptor having been written to the second one of the doorbell addresses, sending the second message from the network interface adapter over the network.

2. A method according to claim 1, wherein assigning the one or more doorbell addresses comprises allocating a priority area for writing the descriptors within an address range defined by the one or more doorbell addresses, and wherein writing the second descriptor comprises writing the second descriptor to the priority area.

3. A method according to claim 2, wherein writing the second descriptor to the priority area comprises writing the second descriptor after writing the command to the first one of the doorbell addresses, and wherein sending the second message comprises, responsive to writing the second descriptor to the priority area, sending the second message before sending the first message.

4. A method according to claim 2, wherein writing the second descriptor comprises writing the second descriptor to the system memory, as well as to the priority area, and wherein sending the second message comprises, when the second descriptor is successfully written in its entirety to the priority area, executing the second descriptor written to the priority area without reading the second descriptor from the system memory.

5. A method according to claim 1, wherein writing the first and second descriptors comprises indicating first and second ranges of data to be read from the system memory for inclusion in the first and second messages, respectively, and wherein sending the first and second messages comprises reading the data from the first and second ranges responsive to the first and second descriptors.

6. A method according to claim 5, wherein reading the data comprises reading the data using direct memory access (DMA) by the network interface adapter to the system memory.

7. A method according to claim 1, wherein assigning the one or more doorbell addresses comprises assigning first and second doorbell addresses

respectively to first and second processes running on the host processor, and wherein writing the command comprises writing the command to the first doorbell address using the first process, and writing the second descriptor comprises writing the second descriptor to the second doorbell address using the second process.

8. A method according to claim 1, wherein sending the first and second messages comprises sending one or more data packets over the network for each of the messages.

9. A method according to claim 8, wherein the network comprises a switch fabric, and wherein the network interface adapter comprises a host channel adapter (HCA), and wherein writing the first and second descriptors comprises submitting work requests (WRs) for execution by the HCA.

10. A method for direct memory access (DMA), comprising:

writing a first descriptor to a system memory associated with a host processor, the first descriptor defining a first operation for execution by a DMA engine;

writing a command to a first doorbell address of the DMA engine, instructing the engine to read and execute the first descriptor;

writing a second descriptor to a second doorbell address of the DMA engine, the second descriptor defining a second operation for execution by the DMA engine;

responsive to the command written to the first doorbell address, reading the first descriptor from the system memory and executing the first descriptor using the DMA engine; and

responsive to the second descriptor having been written to the second doorbell address, executing the second descriptor using the DMA engine.

11. A method according to claim 10, wherein writing the second descriptor comprises writing the second descriptor to a priority area allocated for writing the descriptors within an address range of the doorbell addresses.

12. A method according to claim 11, wherein writing the second descriptor to the priority area comprises writing the second descriptor after writing the command to the first doorbell address, and wherein executing the second descriptor comprises, responsive to writing the second descriptor to the priority area, executing the second descriptor before executing the first descriptor.

13. A method according to claim 11, wherein writing the second descriptor comprises writing the second descriptor to the system memory, as well as to the priority area, and wherein executing the second descriptor comprises, when the second descriptor is successfully written in its entirety to the priority area, reading and executing the second descriptor written to the priority area using the DMA engine, without reading the second descriptor from the system memory.

14. A method according to claim 10, wherein writing the first and second descriptors comprises indicating first and second address ranges, respectively, in the system memory, and wherein executing the first and second descriptors comprises at least one of a scatter step, comprising conveying data from a data source to at least

one of the first and second address ranges, and a gather step, comprising conveying data from at least one of the first and second address ranges to a data target.

15. A network interface adapter, for coupling a host processor to a communication network, the adapter comprising:

a range of doorbell addresses in an address space of the host processor, the range including first and second doorbell addresses;

execution circuitry, adapted to send messages over the network responsive to descriptors prepared by the host processor, the descriptors including first and second descriptors; and

a doorbell handler, which is coupled to the range of doorbell addresses so as to receive a command written by the host processor to the first doorbell address, indicating that the first descriptor has been written to a system memory associated with the host processor, the first descriptor defining a first one of the messages, and so as to receive the second descriptor written by the host processor to the second doorbell address, the second descriptor defining a second one of the messages, the doorbell handler being further coupled, responsive to the command having been written to the first doorbell address, to instruct the execution circuitry to read the first descriptor from the system memory and to execute the first descriptor so as to send the first one of the messages, and responsive to the second descriptor having been written to the second doorbell address, to pass the second descriptor to the execution circuitry and to instruct the execution circuitry to execute the second descriptor so as to send the second one of the messages.

16. An adapter according to claim 15, wherein the second doorbell address is in a priority area within the address range, allocated for writing the descriptors thereto by the host processor.

17. An adapter according to claim 16, wherein the execution circuitry comprises a scheduler, which is adapted to determine an order of execution of the descriptors by the execution circuitry, and wherein responsive to the second descriptor having been written to the priority area, the doorbell handler is adapted to place the second descriptor in the order for execution ahead of the first descriptor.

18. An adapter according to claim 16, wherein the second descriptor is written by the host processor to the system memory, as well as to the priority area, and wherein the doorbell handler is adapted, when the second descriptor is successfully written in its entirety to the priority area, to pass the second descriptor to the execution circuitry without instructing the execution circuitry to read the second descriptor from the system memory.

19. An adapter according to claim 15, wherein the first and second descriptors indicate first and second ranges of data to be read from the system memory for inclusion in the first and second messages, respectively, and wherein the execution circuitry is adapted to read the data from the first and second ranges responsive to the first and second descriptors.

20. An adapter according to claim 19, wherein the execution circuitry comprises a gather engine, which is coupled to read the data by direct memory access (DMA) to the system memory.

21. An adapter according to claim 15, wherein the first and second doorbell addresses are assigned respectively to first and second processes running on the host processor, and wherein the command is written to the first doorbell address using the first process, and the second descriptor is written to the second doorbell address using the second process.

22. An adapter according to claim 15, wherein the execution circuitry is adapted to send the first and second messages by generating data packets to send over the network for each of the messages.

23. An adapter according to claim 22, wherein the network comprises a switch fabric, and wherein the network interface adapter comprises a host channel adapter (HCA), and wherein the first and second descriptors comprise work requests (WRs) submitted by the host processor for execution by the HCA.

24. A host channel adapter, for coupling a host processor to a switch fabric, the adapter comprising:

a range of doorbell addresses in an address space of the host processor, the range including first and second doorbell addresses;

execution circuitry, adapted to generate data packets for transmission over the network responsive to work requests prepared by the host processor, the work requests including first and second work requests; and

a doorbell handler, which is coupled to the range of doorbell addresses so as to receive a command written by the host processor to the first doorbell address, indicating that the first work request has been written to a system memory associated with the host processor, and so as to receive the second work request written by the host processor to the second doorbell address, the doorbell handler being further coupled, responsive to the command having been written to the first doorbell address, to pass instructions to the execution circuitry to read the first work request from the system memory and to execute a first work queue element corresponding to the first work request so as to generate the data packets called for by the first work request, and responsive to the second work request having been written to the second doorbell address, to pass a work queue element corresponding to the second work request to the execution circuitry and to instruct the execution circuitry to execute the second work queue element so as to generate the data packets called for by the second work request.

25. A direct memory access (DMA) device, comprising:

a range of doorbell addresses in an address space of a host processor, the range including first and second doorbell addresses;

a DMA engine, adapted to access a system memory associated with the host processor, responsive to descriptors prepared by the host processor, the descriptors including first and second descriptors defining respective first and second operations for execution by the DMA engine; and

a doorbell handler, which is coupled to the range of doorbell addresses so as to receive a command written by the host processor to the first doorbell address, indicating that the first descriptor has been written to the system memory, and so as to receive the second descriptor written by the host processor to the second doorbell address, the doorbell handler being further coupled, responsive to the command having been written to the first doorbell address, to instruct the DMA engine to execute the first operation responsive to the first descriptor in the system memory, and responsive to the second descriptor having been written to the second doorbell address, to instruct the DMA engine to execute the second operation.

26. A device according to claim 25, wherein the second doorbell address is in a priority area within the address range, allocated for writing the descriptors thereto by the host processor.

27. A device according to claim 26, and comprising a scheduler, which is adapted to determine an order of execution of the operations by the DMA engine, and wherein responsive to the second descriptor having been written to the priority area, the doorbell handler is adapted to place the second operation in the order for execution ahead of the first operation.

28. A device according to claim 26, wherein the second descriptor is written by the host processor to the system memory, as well as to the priority area, and wherein the doorbell handler is adapted, when the second descriptor is successfully written in its entirety to the priority area, to pass the second descriptor to

the DMA engine for execution without reading the second descriptor from the system memory.

29. A device according to claim 25, wherein the first and second descriptors indicate first and second address ranges, respectively, in the system memory, and wherein the first and second operations executed by the DMA engine comprise at least one of a scatter operation, comprising conveying data from a data source to at least one of the first and second address ranges, and a gather operation, comprising conveying data from at least one of the first and second address ranges to a data target.

IX. APPENDIX OF EVIDENCE

NONE

X. APPENDIX OF RELATED PROCEEDINGS

NONE